UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/065,378	10/10/2002	Sheng-Chung Wu	JCLA6435	1015

23900 7590 11/29/2004

J C PATENTS, INC.
4 VENTURE, SUITE 250
IRVINE, CA 92618

EXAMINER	
TRUONG, BAO Q	
ART UNIT	**PAPER NUMBER**
2187	

DATE MAILED: 11/29/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/065,378	WU ET AL.	
	Examiner	Art Unit	
	Bao Q Truong	2187	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _23 August 2004_.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-14_ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☒ Claim(s) _14_ is/are allowed.

6)☒ Claim(s) _1,2,6-8,12 and 13_ is/are rejected.

7)☒ Claim(s) _3-5 and 9-11_ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _10 October 2002_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

Response to Amendment

1. The examiner acknowledges the applicant's submission of Amendment for Application No. 10/065,378 dated on 15 September 2004. At this point, claims 8 and 13 have been amended; claim 14 has been added. There are 14 claims pending in the application; there are 4 independent claims and 10 dependent claims, all of which are ready for reconsideration by the examiner.

Claim Rejections - 35 USC § 102

2. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

3. Claims 1-2, 6-8, and 12-13 are rejected under 35 U.S.C. 102(b) as being anticipated by Young (U.S. Patent No. 5,991,819).

Referring to claim 1, Young teaches a method of operating a control chip having a multiple-layer defer queue between a first bus and a second bus (see figure 3-4), the method comprising:

storing a request in the multiple-layer defer queue, wherein the request is issued by the first bus as receiving requests from a CPU and storing the requests in a Pending Buffer (see figure 4: element 435; and column 9: lines 15-52);

issuing a defer response or a retry response with respect to the request to the first bus as loading the requests into a Pending Buffer to be deferred or retried (see figure 4: elements 417, 435, 437; column 9: lines 42-52; column 16: lines 37-67; and column 17: lines 1-31);

issuing the request to the second bus as sending the requests from the I/O Inbound Request Queue to a PCI device couple to an I/O bus (see figure 3: elements 309A-B; and column 14: lines 47-53);

receiving a responded data with respect to the request from the second bus as receiving data with respect to the requests from the I/O bus at an Outbound Data Buffer (see figure 5E: element 511; and column 14: lines 57-63);

providing the responded data to the first bus if the defer response issues to the first bus as providing data in response to the requests (see figure 5F: element 503; and column 12: lines 62-67); and

providing the responded data to the first bus if the retry response issues to the first bus and only when the first bus again issues the request as in response to the requests that are loaded

into the Pending Buffer to be retried, providing data after the requests are retried (see column 17: lines 19-31).

As to claim 2, Young further teaches that, in the step of storing the request into the multiple-layer defer queue, the multiple-layer defer queue also provides a request record with respect to the request as providing an address/command record (see column 9: lines 27-41).

As to claim 6, Young further teaches that the requests includes one selected from the group consisting of at least one input/output (I/O) read request, at least one I/O write request, and at least one memory read request as CPU read/write requests to PCI device (see figures 3 and 4).

Referring to claim 7, Young teaches a method of operating a control chip having a multiple-layer defer queue between a first bus and a second bus (see figures 3-4), the method comprising:

storing a plurality of requests issued from the first bus in the multiple-layer defer queue as receiving requests from a CPU and storing the requests in a Pending Buffer (see figure 4: element 435; and column 9: lines 15-52), wherein each of the requests has corresponding one response selected from the group consisting of a retry response and a defer response to be responded to the first bus as loading the requests into a Pending Buffer to be deferred or retried (see figure 4: elements 417, 435, 437; column 9: lines 42-51; column 16: lines 37-67; and column 17: lines 1-31);

sequentially issuing the requests to the second bus, wherein the requests at least includes a first request as sending the requests from the I/O Inbound Request Queue to a PCI device couple to an I/O bus (see figure 3: elements 309A-B; and column 14: lines 47-53);

receiving a responded data with respect to the first request from the second bus as receiving data with respect to the requests from the I/O bus at an Outbound Data Buffer (see figure 5E: element 511; and column 14: lines 57-63);

providing the responded data to the first bus if the defer response with respect to the first request issues to the first bus as providing data in response to the requests (see figure 5F: element 503; and column 12: lines 62-67); and

providing the responded data to the first bus if the retry response with respect to the first request issues to the first bus and only when the first bus again issues the first request as in response to the requests that are loaded into the Pending Buffer to be retried, providing data after the requests are retried (see column 17: lines 19-31).

As to claim 8, Young further teaches that, in the step of storing the requests issued from the first bus into the multiple-layer defer queue, the multiple-layer defer queue also provides a request record with respect to the request as providing an address/command record (see column 9: lines 27-41).

As to claim 12, Young further teaches that the requests includes one selected from the group consisting of at least one input/output (I/O) read request, at least one I/O write request, and at least one memory read request as CPU read/write requests to PCI device (see figures 3 and 4).

Referring to claim 13, Young discloses a control chip with multi-layer defer queue (see figures 3-4), coupled to a CPU bus and a PCI bus, the control chip comprising:

a PCI request queue, receiving a CPU request from the CPU bus, and generating a PCI request record (see figure 4: element 405; and column 9: lines 15-40);

a multi-layer defer queue, when receiving the CPU request, respectively responding to the CPU bus by one of a defer response and a retry response as receiving requests from a CPU and storing the requests in a Pending Buffer (see figure 4: element 435; and column 9: lines 15-52); loading the requests into a Pending Buffer to be deferred or retried (see figure 4: elements 417, 435, 437; column 9: lines 42-51; column 16: lines 37-67; and column 17: lines 1-31);

a PCI access queue, receiving the PCI request record (see figure 4: element 417; and column 14: lines 47-53); and

a PCI controller, receiving the request from the multi-layer defer queue, causing the PCI request record of the PCI access queue to be transmitted to the PCI bus via the PO controller (see figure 4: element 413; column 13: lines 40-67; and column 14: lines 1-12);

wherein when the PCI bus generates a response data and if the CPU request in the multi-layer defer queue is to produce the defer response, then the response data is directly sent to CPU bus as providing data in response to the requests (see figure 5F: element 503; and column 12: lines 62-67), if the CPU request in the multi-layer defer queue is to produce the retry response and the CPU bus issues the CPU request, then the response data is transmitted to the CPU bus as in response to the requests that are loaded into the Pending Buffer to be retried, providing data after the requests are retried (see column 17: lines 19-31).

4. Claims 1-2, 6-8, and 12-13 are rejected under 35 U.S.C. 102(e) as being anticipated by Chin et al. (U.S. Patent No. 6,247,102 B1).

Referring to claim 1, Chin teaches a method of operating a control chip having a multiple-layer defer queue between a first bus and a second bus, the method comprising:

storing a request in the multiple-layer defer queue, wherein the request is issued by the first bus as a deferred queue storing CPU-to-PCI transaction cycles (see figure 3: elements 142, 144, 146; column 7: lines 64-67; column 8: lines 1-34);

issuing a defer response or a retry response with respect to the request to the first bus as a deferred queue responding to CPU-to-PCI transaction cycles with either defer response or non-defer (retry) response (see column 30: lines 57-67; and column 31: lines 1-12);

issuing the request to the second bus as sending the CPU-to-PCI transaction cycles to a PCI interface for placing on a PCI device (see figure 2: element 160; and figure 5);

receiving a responded data with respect to the request from the second bus as receiving data with respect to the CPU-to-PCI transaction cycles (see column 22: lines 23-46);

providing the responded data to the first bus if the defer response issues to the first bus; and providing the responded data to the first bus if the retry response issues to the first bus and only when the first bus again issues the request as either deferring a transaction or retrying a transaction cycle (see figures 12; column 22: lines 23-46; column 30: lines 57-67; and column 31: lines 1-12).

As to claim 2, Chin further teaches that, in the step of storing the requests into the multiple-layer defer queue, the multiple-layer defer queue also provides a request record with respect to the request as an in-order queue having a pointer to keep track of transaction cycles (see figure 3: elements 133, 138; column 7: lines 26-28, lines 43-47; and column 8: lines 3-5).

As to claim 6, Chin further teaches that the request includes one selected from the group consisting of an input/output (I/O) read request, an I/O write request, and a memory read request as CPU-to-PCI read/write transactions (see figures 11 and 12).

Referring to claim 7, Chin teaches a method of operating a control chip having a multiple-layer defer queue between a first bus and a second bus, the method comprising:

storing a plurality of requests issued from the first bus in the multiple-layer defer queue as a deferred queue storing CPU-to-PCI transaction cycles (see figure 3: elements 142, 144, 146; column 7: lines 64-67; column 8: lines 1-34), wherein each of the requests has corresponding one response selected from the group consisting of a retry response and a defer response to be responded to the first bus as a deferred queue responding to CPU-to-PCI transaction cycles with either defer response or non-defer (retry) response (see column 30: lines 57-67; and column 31: lines 1-12);

sequentially issuing the requests to the second bus, wherein the requests at least includes a first request as sending the CPU-to-PCI transaction cycles to a PCI interface for placing on a PCI device (see figure 2: element 160; and figure 5);

receiving a responded data with respect to the first request from the second bus as receiving data with respect to the CPU-to-PCI transaction cycles (see column 22: lines 23-46);

providing the responded data to the first bus if the defer response with respect to the first request issues to the first bus and providing the responded data to the first bus if the retry response with respect to the first request issues to the first bus and only when the first bus again issues the first request as either deferring a transaction or retrying a transaction cycle (see figures 12; column 22: lines 23-46; column 30: lines 57-67; and column 31: lines 1-12).

As to claim 8, Chin further teaches that, in the step of storing the requests issued from the first bus into the multiple-layer defer queue, the multiple-layer defer queue also provides a request record with respect to the request as an in-order queue having a pointer to keep track of transaction cycles (see figure 3: elements 133, 138; column 7: lines 26-28, lines 43-47; and column 8: lines 3-5).

As to claim 12, Chin further teaches that the requests includes one selected from the group consisting of at least one input/output (I/O) read request, at least one I/O write request, and at least one memory read request as CPU-to-PCI read/write transactions (see figures 11 and 12).

Referring to claim 13, Chin discloses a control chip with multi-layer defer queue, coupled to a CPU bus and a PCI bus, the control chip comprising:

a PCI request queue, receiving a CPU request from the CPU bus, and generating a PCI request record as a PCI request queue storing CPU to PCI transactions (see figure 3: element 134; and column 7: lines 36-63);

a multi-layer defer queue, when receiving the CPU request, respectively responding to the CPU bus by one of a defer response and a retry response as a deferred queue responding to CPU transactions with either defer response or non-defer (retry) response (see figure 3: elements 142, 144, 146; column 7: lines 64-67; column 8: lines 1-34; column 30: lines 57-67; and column 31: lines 1-12);

a PCI access queue, receiving the PCI request record as a CPU to PCI queue (see figure 2: element 184); and

a PCI controller, receiving the request from the multi-layer defer queue, causing the PCI request record of the PCI access queue to be transmitted to the PCI bus via the PO controller as a PCI interface (see figure 2: element 16; and figure 5);

wherein when the PCI bus generates a response data and if the CPU request in the multi-layer defer queue is to produce the defer response, then the response data is directly sent to CPU bus, if the CPU request in the multi-layer defer queue is to produce the retry response and the CPU bus issues the CPU request, then the response data is transmitted to the CPU bus as either deferring a transaction or retrying a transaction (see figures 12; column 22: lines 23-46; column 30: lines 57-67; and column 31: lines 1-12).

Response to Arguments

5. Applicant's arguments, filed on 15 September 2004, with respect to claims 1, 7, and 13 over Young have been fully considered but they are not persuasive.

With respect to claim 1, the applicant argues on page 10 that Young does not teach "issuing a defer response or a retry response with respect to the request to the first bus" as in claim 1. The examiner disagrees and directs the applicant, once again, to figure 4: element 435. Element 435 is a pending buffer which stores pending transaction (see column 16: lines 30-37). In figure 4, an output signal from the pending buffer 435 is compared with a signal from the processor bus 330A to generate a retry logic signal.

With respect to claim 1, the applicant argues on page 10 that Young does not teach "providing the responded data to the first bus if the retry response issues to the first bus; and providing the responded data to the first bus if the retry response issues to the first bus and only when the first bus again issues the request" as in claim 1. The examiner disagrees and directs the applicant, once again, to figure 5F: element 503. Element 503 is an inbound data buffer used to hold read data for processor read transaction included deferred transactions (see column 12: lines 62-67).

With respect to claim 7, the applicant argues on page 10 that Young does not teach "storing a plurality of requests issued from the first bus in the multiple-layer defer queue, wherein each of the requests has corresponding one response selected from the group consisting of a retry response and a defer response to be responded to the first bus" as in claim 7. The

examiner disagrees and directs the applicant, once again, to response to applicant's argument with respect to claim 1 above. In figure 4, element 435 is a pending buffer which stores pending transaction (see column 16: lines 30-37). In figure 4, an output signal from the pending buffer 435 is compared with a signal from the processor bus 330A to generate a retry logic signal.

With respect to claim 7, the applicant argues on page 10 that Young does not teach "providing the responded data to the first bus if the retry response issues to the first bus; and providing the responded data to the first bus if the retry response issues to the first bus and only when the first bus again issues the request" as in claim 7. The examiner disagrees and directs the applicant, once again, to response to applicant's argument with respect to claim 1 above. In figure 4, an output signal from the pending buffer 435 is compared with a signal from the processor bus 330A to generate a retry logic signal. In figure 5F, element 503 is an inbound data buffer used to hold read data for processor read transaction **included deferred transactions** (see column 12: lines 62-67).

With respect to claim 13, the applicant argues on page 10 that Young does not disclose "a PCI request queue for receiving a CPU request from the CPU bus, and generating a PCI request record" as in claim 13. The examiner disagrees and directs the applicant, once again, to figure 4: element 405 and column 9: lines 27-34. Element 405 in figure 4 is an outbound request queue to hold pending CPU local and remote requests (see column 9: lines 23-25). CPU remote requests target I/O devices (PCI devices are I/O devices) (see column 9: lines 45-47). Element 405 also contains logical records for each pending request (see column 9: lines 27-34).

With respect to claim 13, the applicant argues on page 10 that Young does not disclose "a multi-layer defer queue, when receiving the CPU request, respectively responding to the CPU bus by one of a defer response and a retry response" as in claim 13. The examiner disagrees and directs the applicant, once again, to response to applicant's argument with respect to claim 1 above. In figure 4, element 435 is a pending buffer which stores pending transaction (see column 16: lines 30-37). In figure 4, an output signal from the pending buffer 435 is compared with a signal from the processor bus 330A to generate a retry logic signal.

With respect to claim 13, the applicant argues on page 10 that Young does not disclose "when the PCI bus generates a response data and if the CPU request in the multi-layer defer queue is to produce the defer response, then the response data is directly sent to the CPU, if the CPU request in the multi-layer defer queue is to produce the retry response and the CPU bus issues the CPU request, then the response data is transmitted to the CPU bus" as in claim 13. The examiner disagrees and directs the applicant, once again, to figure 5F: element 503; column 12: lines 62-67; and column 17: lines 19-31. In figure 5F, element 503 is an inbound data buffer used to hold read data for processor read transaction **included deferred transactions** (see column 12: lines 62-67).

6. Applicant's arguments, filed on 15 September 2004, with respect to claims 1, 7, and 13 over Chin have been fully considered but they are not persuasive.

With respect to claim 1, the applicant argues on page 15 that Chin does not teach "storing a request in the multiple-layer defer queue, wherein the request is issued by the first bus" as in claim 1. The examiner disagrees and directs the applicant, once again, to figure 3: element144. The examiner further directs the applicant to figure 3: element 132 and column 7: lines 36-47. In figure 3, all CPU bus cycles targeted for either the memory controller, AGP interface, or PCI interface are loaded into the in-order-queue 132 (see column 7: lines 41-43). CPU bus cycles that target the PCI interface can be deferred to a deferred queue (see figure 3: element 144).

With respect to claim 1, the applicant argues on page 15 that Chin does not teach "providing the responded data to the first bus if the retry response issues to the first bus; and providing the responded data to the first bus if the retry response issues to the first bus and only when the first bus again issues the request" as in claim 1. The examiner disagrees and directs the applicant, once again, to column 22: lines 23-46 and column 31: lines 1-12. A CPU read request to the PCI interface can be either deferred or non-deferred. In either case, data read from PCI bus is returned to the CPU.

With respect to claim 7, the applicant argues on page 15 that Chin does not teach "storing a plurality of requests issued from the first bus in the multiple-layer defer queue, wherein each of the requests has corresponding one response selected from the group consisting of a retry response and a defer response to be responded to the first bus" as in claim 7. The examiner

disagrees and directs the applicant, once again, to figure 3: element 132 and column 7: lines 36-47. In figure 3, all CPU bus cycles targeted for either the memory controller, AGP interface, or PCI interface are loaded into the in-order-queue 132 (see column 7: lines 41-43). CPU bus cycles that target the PCI interface can be deferred to a deferred queue (see figure 3: element 144). A CPU read request to the PCI interface can be either deferred or non-deferred (see column 31: lines 1-12).

With respect to claim 7, the applicant argues on page 15 that Chin does not teach "providing the responded data to the first bus if the retry response issues to the first bus; and providing the responded data to the first bus if the retry response issues to the first bus and only when the first bus again issues the request" as in claim 7. The examiner disagrees and directs the applicant, once again, to column 22: lines 23-46 and column 31: lines 1-12. A CPU read request to the PCI interface can be either deferred or non-deferred. In either case, data read from PCI bus is returned to the CPU.

With respect to claim 13, the applicant argues on page 15 that Chin does not disclose "a PCI request queue for receiving a CPU request from the CPU bus, and generating a PCI request record" as in claim 13. The examiner disagrees and directs the applicant, once again, to figure 3: elements 132, 134; column 7: lines 28-31, lines 41-43. CPU cycles targeted PCI interface are placed in a PCI request queue and an in-order-queue. Furthermore, a pointer is provided to keep record of transaction cycles (see column 7: lines 43-47 and column 8: lines 3-5.

With respect to claim 13, the applicant argues on page 15 that Chin does not disclose "a multi-layer defer queue, when receiving the CPU request, respectively responding to the CPU bus by one of a defer response and a retry response" as in claim 13. The examiner disagrees and directs the applicant, once again, to figure 3: element 132 and column 7: lines 36-47. In figure 3, all CPU bus cycles targeted for either the memory controller, AGP interface, or PCI interface are loaded into the in-order-queue 132 (see column 7: lines 41-43). CPU bus cycles that target the PCI interface can be deferred to a deferred queue (see figure 3: element 144). A CPU read request to the PCI interface can be either deferred or non-deferred (see column 31: lines 1-12).

With respect to claim 13, the applicant argues on page 15 that Chin does not disclose "when the PCI bus generates a response data and if the CPU request in the multi-layer defer queue is to produce the defer response, then the response data is directly sent to the CPU, if the CPU request in the multi-layer defer queue is to produce the retry response and the CPU bus issues the CPU request, then the response data is transmitted to the CPU bus" as in claim 13. The examiner disagrees and directs the applicant, once again, to column 22: lines 23-46 and column 31: lines 1-12. A CPU read request to the PCI interface can be either deferred or non-deferred. In either case, data read from PCI bus is returned to the CPU.

Allowable Subject Matter

7. Claims 3-5 and 9-11 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

8. Claim 14 is allowed.

Conclusion

9. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **three months** from the mailing date of this action. In the event a first reply is filed within **two months** of the mailing date of this final action and the advisory action is not mailed until after the end of the **three-month** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **six months** from the mailing date of this final action.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Bao Q Truong whose telephone number is (571) 272-4202. The examiner can normally be reached on Monday-Friday from 6:00 AM to 3:00 PM (ET).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Donald A Sparks, can be reached on (571) 272-4201. The fax phone number for the organization where this application or proceeding is assigned is (703) 746-7239.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (571) 272-2100.

BAO QUOC TRUONG

BT

Patent Examiner

22 November 2004

Donald A. Sparks

Supervisory Patent Examiner

Technology Center 2100